



A Framework for the Control of a Parallel Manipulator with Several Actuation Modes

Damien Chablat, Ranjan Jha, Stéphane Caro

► To cite this version:

Damien Chablat, Ranjan Jha, Stéphane Caro. A Framework for the Control of a Parallel Manipulator with Several Actuation Modes. 14th International Conference on Industrial Informatics, Jul 2016, Poitiers, France. hal-01313376

HAL Id: hal-01313376

<https://hal.science/hal-01313376>

Submitted on 9 May 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Framework for the Control of a Parallel Manipulator with Several Actuation Modes

Damien Chablat, Ranjan Jha, Stéphane Caro

Institut de Recherche en Communications et Cybernétique de Nantes

UMR CNRS 6597, Nantes, France

Email: Damien.Chablat@cnrs.fr, Ranjan.Jha@irccyn.ec-nantes.fr, Stephane.Caro@irccyn.ec-nantes.fr

Abstract—There have been several research works on reconfigurable parallel manipulators in the last few years. Some robots are reconfigurable in the sense that the position of the anchor points on the moving platform or the actuated joints can be changed. Some problems may arise when one intends to make a prototype and develop its control scheme. A reconfigurable planar parallel robot, named NaVARo, is a 3-DOF planar parallel manipulator with eight actuation modes. The subject of this paper is about a control scheme of NaVARo while taking advantage of multiple sensors such as motor encoders, additional absolute encoders and magnetic sensors used to determine the current assembly mode of the manipulator. Finally, a methodology is presented to determine the home configuration of NaVARo.

I. INTRODUCTION

There are many types of reconfigurable parallel manipulators. This paper deals with the determination of a control scheme for a reconfigurable parallel manipulator with different actuation schemes. It is well known that the kinetostatic performance of parallel manipulators may change drastically in the neighborhood of parallel singularities. Parallel mechanisms may have several solutions to their inverse kinematics problem, i.e. several working modes. It is usually difficult to get a large singularity-free workspace for a given working mode. Hence, one must plan the trajectory in such a way that there should be change of working mode to avoid the singularities. These problems can be solved by introducing actuation redundancy, which involves force control algorithms [1]. The kinetostatic performance of a planar parallel mechanism, NaVARo, with variable actuation modes is presented in [2], [3]. The homing, calibration and model-based predictive control for the 4RPR parallel architecture is shown in [4].

In [5], a 3-CRU reconfigurable parallel manipulator is presented, which can work in two different modes, the first mode gives three degrees of freedom of pure translation motion and another of pure rotation. Moreover, it is shown that there is a possibility of switching between the two modes by simultaneous change of configuration of all the universal joints. The reconfiguration analysis of a 2-DOF 3-RRRR parallel mechanism is presented in [6], which has six operation modes, two spherical translational modes, two planar motion modes and two sphere-on-sphere rolling modes. A reconfigurable parallel mechanism with planar five bar metamorphic linkages is presented in [7]. A self-calibration method, which is based on leg-end distance errors and measurement residues for the three-legged modular reconfigurable

parallel robots are presented in [8] and [9], respectively. In [10] a metamorphic parallel mechanism is presented which can switch its motion from a 3-dof translational motion to a 3-dof orientation motion. This paper presents the actuation modes of a variable actuation parallel manipulator, NaVARo. Besides, a framework is presented to follow any trajectory using its inverse geometric model only.

The paper is organized as follows. We first introduce the NaVARo, a 3-RRR parallel robot with eight actuation modes. We then present the direct kinematics of 3RPR and 3RRR parallel mechanisms with the joint space and workspace analysis for these mechanisms. In later sections, homing procedure and motions associated with the NaVARo robot, which includes inverse kinematics and Jacobian matrix computations are presented.

II. THE NAVARo ROBOT

A. Robot under study

The NaVARo robot was introduced in [2] and illustrated in Fig. 1. From the classical 3-RRR parallel robot, a parallelogram is added to each leg and connected to the first limb. A virtual model of NaVARo robot is shown in Fig. 2 which is used for analyzing the different aspects associated with the robot under study. Virtually, the actuator positions can change from the revolute joint located on A_i or B_i . This feature exists thanks to a clever transmission system made with two clutches. The NaVARo robot admits eight actuation modes if we assume that only one clutch can be activated at the same time per leg.



Fig. 1: The NaVARo robot

The dimensions were chosen so that the forward kinematics of the 3-RPR robot associated can be solved analytically. Thus, the points A_i and C_i are the vertices of two equilateral triangles, respectively, of side lengths 0.7 m and 0.35 m, respectively. The lengths of the links A_iB_i and B_iC_i are equal to 0.21 m for $i = 1, 2, 3$. Due to the parallelograms, the minimum and maximum distances between points A_i and C_i are 0.08 m and 0.039 m, respectively.

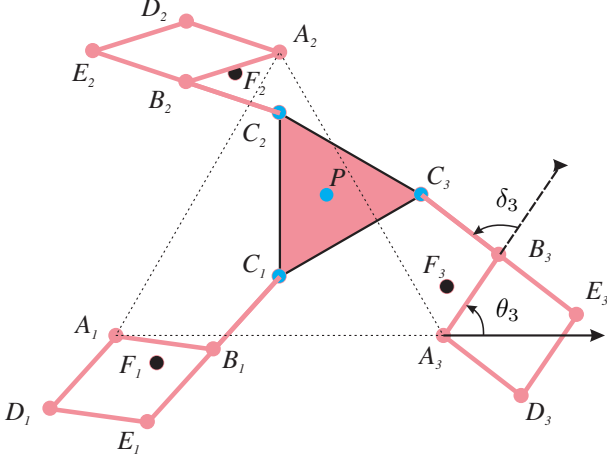


Fig. 2: The virtual model of the NaVARo robot

B. Transmission System

A transmission system has been developed and mounted in each leg of the NaVARo in order for the manipulator to be able to switch smoothly from one actuation mode to another along a prescribed trajectory. Figure 3 illustrates a CAD modeling of the transmission system of the NaVARo. This system can be seen as a double clutch system and contains: (i) a motor; (ii) a gearhead, (iii) a motor shaft, (iv) a main shaft (in cyan), (v) a base (in yellow), (vi) a housing (in purple) and (vii) two clutches (in brown). As a matter of fact, the two clutches 1 and 2 are electromagnetic brakes.

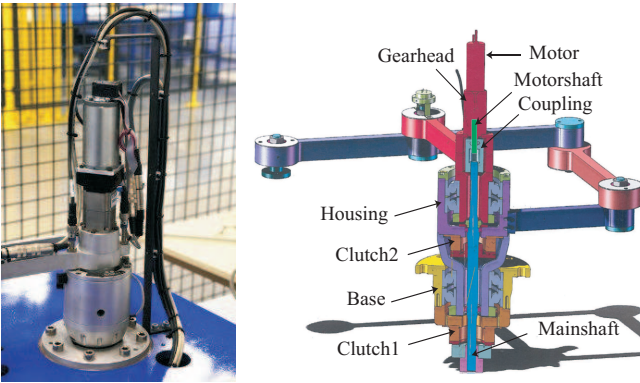


Fig. 3: The NaVARo transmission system

Each transmission system has four actuation schemes that are defined thereafter [12]:

- 1) None of clutches 1 and 2 are active: The main shaft is free to move with respect to the housing and the base. In that case, none of the first two revolute joints of the corresponding legs are actuated, namely, angles θ_i and δ_i are passive, $i = 1, 2, 3$.
- 2) Clutch 1 is active while Clutch 2 is not: The main shaft is fixed with respect to the base, i.e., the link A_iB_i is driven thanks to the rotation of the motor shaft. In that case, angle θ_i is active and angle δ_i is passive.
- 3) Clutch 2 is active while Clutch 1 is not: The main shaft is attached to the housing, but is free to move with respect to the base. In that case, angle θ_i is passive and angle δ_i is active, $i = 1, 2, 3$.
- 4) Both clutches 1 and 2: The blue shaft is attached to both the base and the housing. It means that the housing cannot move and link A_iD_i is fixed. In that case, link C_iE_i performs a circular translation with respect to point A_i . This actuation scheme amounts to an actuated II joint.

Only the second and third actuation schemes of each transmission system are used in the NaVARo prototype in order to keep the three degrees of freedom motion of the moving-platform and to avoid any actuation redundancy and under-actuation.

The fourth actuation mode is used to lock the robot during the actuation mode changing or during the data transfer phases.

TABLE I: The eight actuation modes of the NaVARo

Actuation Mode Number	Driven Links	Active Links
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1B_1, A_2B_2, A_3B_3	$\theta_1, \theta_2, \theta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1B_1, A_2B_2, A_3E_3	$\theta_1, \theta_2, \delta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1B_1, A_2E_2, A_3B_3	$\theta_1, \delta_2, \theta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1E_1, A_2B_2, A_3B_3	$\delta_1, \theta_2, \alpha_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1B_1, A_2E_2, A_3E_3	$\theta_1, \delta_2, \delta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1E_1, A_2E_2, A_3B_3	$\delta_1, \delta_2, \theta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1E_1, A_2B_2, A_3E_3	$\delta_1, \theta_2, \delta_3$
$\underline{RRR}_1 - \underline{RRR}_2 - \underline{RRR}_3$	A_1E_1, A_2E_2, A_3E_3	$\delta_1, \delta_2, \delta_3$

C. Control Hardware and Software

The Control hardware of the NaVARo robot is a 1103 Dspace card [13] with a 450 Mhz Power PC. The actuator positions are required with a frequency equal to 6 kHz and a 200 Hz low pass filter is used to compute the actuator velocity. The robot motions are controlled thanks to a sub program working at 1 kHz. The state of the program is determined as a function of the set of actuators used.

The dynamic effect is added to the robot control scheme to improve the classical PID control scheme. The motor torques Γ_i are generated by removing the coupling between the legs as,

$$\Gamma_i = J \left(\ddot{\theta} + K_P (\theta_i^d - \theta_i) + K_D (\dot{\theta}_i^d - \dot{\theta}_i) + K_I \int_{t_0}^t (\theta_i^d - \theta_i) \right) \quad (1)$$

where $J = 0.7k.m^2$, $\omega = 14 \text{ rad.s}$, $K_P = 3\omega^2$, $K_D = 3\omega$ and $K_I = \omega^3$.

D. Sensor Placements

The NaVARo robot is equipped with three types of sensor. Incremental sensors are located on the shaft of the motors. Analog angle sensors are placed on the revolute joints B_i and C_i to have an absolute measure of these angles. The sensors placed on the pivots B_i allow us to have a measure of the distance $\|A_i C_i\|$ that is equivalent to consider a robot 3-RPR. Magnetic sensors located at F_i are used for the homing robot and at A_i to set the fine leg strokes to prevent the parallelogram is completely squashed or stretched. The location of the magnetic sensors will be described in the Homing Section.

E. Graphic user interface

A graphic user interface (GUI) was developed in Matlab to make first the homing and after the motions (Fig. 4). When the start conditions are satisfied, the homing function can be performed (Section III) and the movements of the robots are made in Cartesian mode (Section IV). The interface displays the value of the actuated and passive joints as well as the current actuation mode. It is also possible to release the servomotors to move the robot by hand.

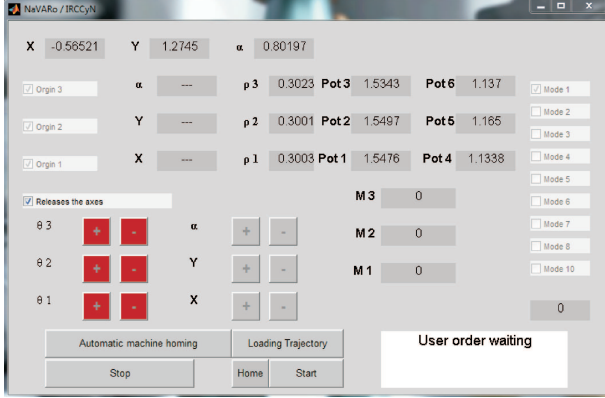


Fig. 4: The graphic user interface of the NaVARo robot

III. HOMING

A. Introduction

The homing procedure permits the robot to reach a pre-defined known position by making a set of actions. The strategy used in our case is to manually place the robot in a given position, checking by sensors the absolute position, then start a movement to accurately detect magnetic sensors. To understand the process, it is necessary to recall some properties of the robot.

B. Direct kinematics of the 3-RPR

When the actuation mode 8 is activated, that is to say, a robot 3-RRR, has direct kinematic model equivalent to that of a robot 3-RPR where $r_i = \|A_i C_i\|$ for $i = 1, 2, 3$. The 3-RPR planar parallel robot with base and mobile platform admits only a maximum of four real solutions to the direct kinematic problem (DKP) [14], shown in Fig. 5(a).

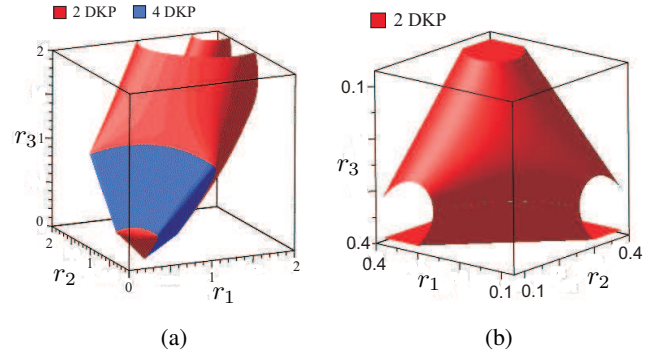


Fig. 5: Joint space of 3RPR parallel mechanism (a) without joint limits (b) with joint limits. The Red and Blue regions represents the two and four real solutions for the DKP, respectively.

The orientation of the mobile platform can be found by solving a second degree polynomial equation in $\cos(\alpha)$:

$$(78400 r_1^2 + 78400 r_2^2 + 78400 r_3^2 - 96040) \cos(\alpha) + 160000 r_1^4 - 160000 r_1^2 r_2^2 - 160000 r_2^2 r_3^2 + 160000 r_2^4 - 160000 r_3^2 r_2^2 + 160000 r_3^4 - 98000 r_1^2 - 98000 r_2^2 - 98000 r_3^2 + 60025 + 38416 \cos(\alpha)^2 = 0 \quad (2)$$

As this quadratic equation depends only on $\cos(\alpha)$, we have a maximum of four solutions. The position x and y can be directly defined as:

$$x = \frac{10 (6 (r_1^2 + r_3^2) - 3C\alpha (r_1^2 - r_3^2) - \sqrt{3}S\alpha (2r_2^2 - r_3^2 - r_1^2))}{105 - 84C\alpha} \quad (3)$$

$$y = \frac{10\sqrt{3}(2(r_1^2 + 2r_2^2 - r_3^2) - S\alpha\sqrt{3}(r_1^2 - r_3^2) - C\alpha(r_1^2 - 2r_2^2 + r_3^2))}{105 - 84C\alpha} \quad (4)$$

where $C\alpha = \cos(\alpha)$ and $S\alpha = \sin(\alpha)$. Due to the joint limits, there are only two real solutions to the direct kinematic problem as shown in Fig. 5(b).

C. Direct kinematics of the 3-RRR

Except for the actuation mode 8, the forward kinematics is general and there are up to 6 real solutions, or assembly modes. Solving this problem amounts to solving a polynomial of degree 6 using the method introduced by Gosselin [14]. It is noteworthy that the polynomial equation cannot be solved in real time on the 1kHz Dspace at hand. However, it is possible to verify the current location by shifting the calculation on the computer controlling the robot. In our case, MATLAB computes all the solutions of the direct kinematics model. However, only the absolute sensors on passive joints distinguish the current assembly mode. In all cases, the resolution is generic and comparable to the resolution of the forward kinematics of the 3-RPR, while considering the length $\|A_i C_i\|$

or $\|B_i C_i\|$ as actuated and knowing the angle δ_i or θ_i respectively.

D. Homing procedure

The home pose lies at the center of the workspace. The proximal link of the legs have to be in front of the magnetic sensor. By knowing the position of this segment is equivalent to study 3-RRR robot with the first motorized joint. To reach this configuration, we can mode the robot in the joint space. We have defined a graphical interface for the displacement of each actuator by addition or subtraction of 5 degrees. At each movement, a trajectory based on a five-degree polynomial is used to have as a desired, zero speed and zero acceleration at the starting point and ending point. Since the algorithm is based on state, a single command can be run simultaneously. When the magnetic sensors are close to the proximal limb, a flag associated with each sensor becomes turned on.

If the robot reaches the limits of movement of parallelograms, security robot is activated and the power of the engine is turned off. It is necessary to move the mobile platform by external effort to depart from this position. For this set of input values, this robot admits two real solutions to the direct geometric model with the same position but two different orientations. As one can not distinguish them by the only position of the proximal, we use sensors placed on the second joint of each leg that has been previously calibrated. For $\theta_1 = -0.585$, $\theta_2 = -2.680$ and $\theta_3 = 1.508$, values for the position of the mobile platform x , y and α_i and the angles γ_i are

- $x = 0$, $y = 0$, $\alpha = 1.0471$ or $\alpha = 0.0414$
- $\delta_1 = \delta_2 = \delta_3 = 1.970$ or $\delta_1 = \delta_2 = \delta_3 = 1.005$

The two assembly modes for the NaVARo robot for the same position of the proximal limbs is shown in Fig. 6.

The homing function is to move from offset each engine separately in the same direction and then return at a constant speed to the sensor. By identifying this position from the home position, we move the robot by joint offset. A flow chart of this algorithm is given in the Fig. 7. When homing is completed, the value encoder is initialized by setting an offset relative to their current values.

The state of the robot is 0 when no movement or command is active. Only the PID keeps the robot in its current position. When the state is not 0, no other command can be initiated. Only security-related commands can interrupt the program. When transferring information associated with a path, the movements are stopped and all clutches are active because this is a priority process. Movement generators are written in C language for generating movements according to the current posture and the end posture.

IV. MOTIONS OF THE ROBOT

To avoid solving the forward kinematics, when the boot process is complete, only Cartesian movements are possible. To change the actuation mode, it is necessary to know how to calculate the inverse kinematics for the first and second

revolute joints of each leg. Similarly, to avoid singular configurations, it is necessary to evaluate the Jacobian matrices by building the position of all segments of the robot.

A. Inverse kinematics model

For changing the actuation mode, the computation of general inverse kinematics model is needed to switch the value of the motor incremental sensor to the current actuated joint. In a first step, we calculate the position of C_i points depending on the position (x, y) and the orientation α of the mobile platform.

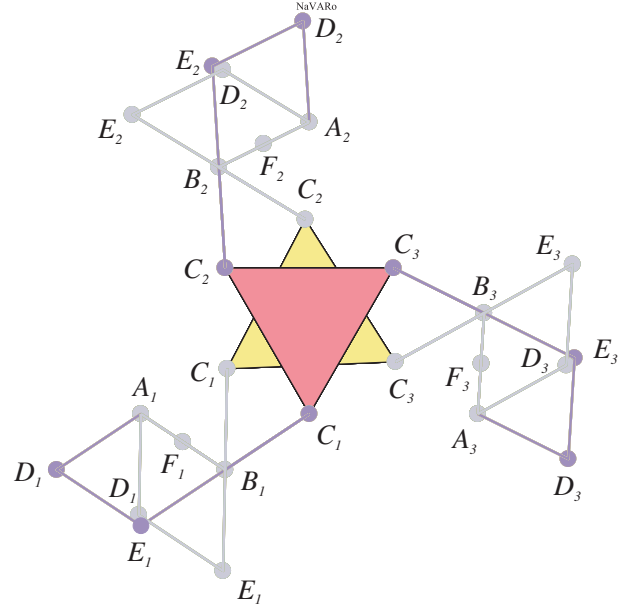


Fig. 6: First assembly mode of the NaVARo shown with pink colored mobile platform and the second assembly mode shown with yellow colored mobile platform for $\theta_1 = -0.585$, $\theta_2 = -2.680$ and $\theta_3 = 1.508$

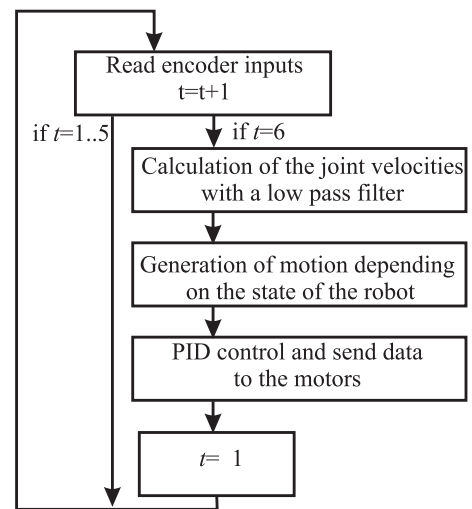


Fig. 7: Flow chart of the control loop

$$\begin{aligned}
C_{1x} &= (x + (C_{1a} \cos(\alpha)) - (C_{1b} \sin(\alpha))), \\
C_{1y} &= (y + (C_{1a} \sin(\alpha)) + (C_{1b} \cos(\alpha))), \\
C_{2x} &= (x + (C_{2a} \cos(\alpha)) - (C_{2b} \sin(\alpha))), \\
C_{2y} &= (y + (C_{2a} \sin(\alpha)) + (C_{2b} \cos(\alpha))), \\
C_{3x} &= (x + (C_{3a} \cos(\alpha)) - (C_{3b} \sin(\alpha))), \\
C_{3y} &= (y + (C_{3a} \sin(\alpha)) + (C_{3b} \cos(\alpha))),
\end{aligned} \quad (5)$$

where C_{ia} and C_{ib} are the x - and y -coordinates of point C_i in the moving frame, respectively. A set of intermediate angles is needed to solve the inverse model that is shown in Fig. 8.

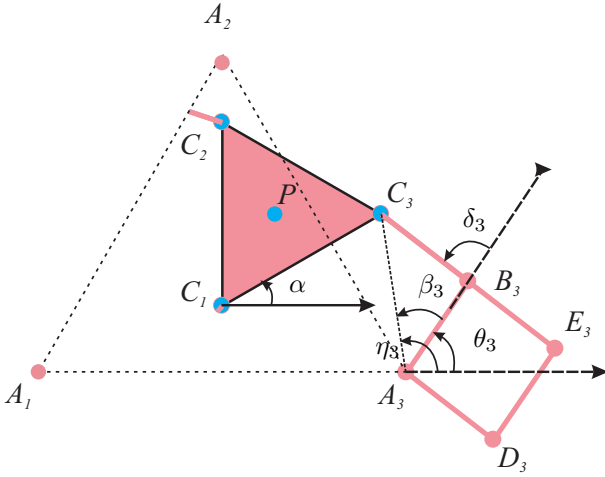


Fig. 8: Definition of the angles used to solve the inverse kinematics

First, we calculate the distance between A_i and C_i .

$$d_i = \sqrt{(C_{1x} - A_{1x})^2 + (C_{1y} - A_{1y})^2} \quad \text{for } i = 1, 2, 3. \quad (6)$$

and by the law of cosines we can determine in the triangles $A_i B_i C_i$

$$\delta_i = \pi \pm \arccos((l_{i1}^2 + l_{i2}^2 - d_i^2)/(2l_{i1}l_{i2})) \quad (7)$$

where l_{i1} and l_{i2} are the proximal and distal lengths

$$\beta_i = \pm \arccos((l_{i1}^2 + d_i^2 - l_{i2}^2)/(2l_{i1}d_i)); \quad (8)$$

and from the position of C_i with respect to A_i

$$\eta_i = \arctan(C_{iy} - A_{iy}, C_{ix} - A_{ix}) \quad (9)$$

Finally, we can evaluate the position of the proximal,

$$\theta_i = \eta_i \pm \beta_i \quad (10)$$

Knowing the current operating mode, the inverse geometric model calculates values of θ_i and δ_i .

B. Jacobian matrices

As NaVARO robot has only one working mode, knowing the position of the mobile platform and using the inverse kinematics, we can know the position of the proximal and distal of the three legs. The kinematic model of NaVARO was introduced in [3]. The following equation defines the Cartesian velocity $\dot{\mathbf{p}}$ of the point P through the three legs.

$$\dot{\mathbf{p}} = \dot{\theta}_i \mathbf{E}(\mathbf{b}_i - \mathbf{a}_i) + \dot{\delta}_i \mathbf{E}(\mathbf{c}_i - \mathbf{b}_i) + \dot{\alpha} \mathbf{E}(\mathbf{p} - \mathbf{c}_i) \quad (11)$$

where

$$\mathbf{E} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (12)$$

and \mathbf{a}_i , \mathbf{b}_i , \mathbf{c}_i and \mathbf{p} are the position vectors of points A_i , B_i , C_i and P and $\dot{\theta}_i$, $\dot{\delta}_i$ and $\dot{\alpha}$ are the joint rates θ_i , δ_i and α as it is shown in Fig. 8.

A generic equation was obtained in [3] to define the Jacobian matrix of NaVARO as

$$\mathbf{A} \mathbf{t} = \mathbf{B} \dot{\mathbf{q}} \quad \text{with} \quad \mathbf{t} = [\dot{\mathbf{p}} \quad \dot{\alpha}]^T \quad \dot{\mathbf{q}} = [\dot{q}_1 \dot{q}_2 \dot{q}_3]^T \quad (13)$$

where $\dot{\mathbf{q}}$ is the joint rate vector which depends on the actuator that is to say the actuated joint, ie $\dot{\theta}_i$ or $\dot{\delta}_i$. The direct and inverse Jacobian matrices are this defined

$$\mathbf{A} = \begin{bmatrix} (\mathbf{c}_1 - \mathbf{h}_1)^T & -(\mathbf{c}_1 - \mathbf{h}_1)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_1) \\ (\mathbf{c}_2 - \mathbf{h}_2)^T & -(\mathbf{c}_2 - \mathbf{h}_2)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_2) \\ (\mathbf{c}_3 - \mathbf{h}_3)^T & -(\mathbf{c}_3 - \mathbf{h}_3)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_3) \end{bmatrix} \quad (14)$$

$$\mathbf{B} = \text{diag} \left[(\mathbf{c}_i - \mathbf{b}_i)^T \mathbf{E}(\mathbf{b}_i - \mathbf{a}_i) \right] \quad \text{for } i = 1, 2, 3 \quad (15)$$

where $\mathbf{h}_i = \mathbf{a}_i$ if the first joint of the leg is actuated and $\mathbf{h}_i = \mathbf{b}_i$ in the other case. To normalize the Jacobian matrix, we use as the characteristic value, the radius r of the mobile platform. Thus the matrix \mathbf{A} becomes

$$\mathbf{A} = \begin{bmatrix} (\mathbf{c}_1 - \mathbf{h}_1)^T & -(\mathbf{c}_1 - \mathbf{h}_1)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_1)/r \\ (\mathbf{c}_2 - \mathbf{h}_2)^T & -(\mathbf{c}_2 - \mathbf{h}_2)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_2)/r \\ (\mathbf{c}_3 - \mathbf{h}_3)^T & -(\mathbf{c}_3 - \mathbf{h}_3)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_3)/r \end{bmatrix} \quad (16)$$

Then, we compute the Jacobian matrix and its condition number based on the Frobenius norm as follows:

$$\mathbf{J} = \mathbf{B}^{-1} \mathbf{A} \quad (17)$$

$$\kappa(\mathbf{J}) = \frac{1}{m} \sqrt{\text{tr}(\mathbf{J}^T \mathbf{J}) \text{tr}(\mathbf{J}^T \mathbf{J})^{-1}} \quad (18)$$

The inverse condition number κ^{-1} is bounded between 0.1 and 1 to avoid any singular configuration during the manipulator motions.

C. Actuation mode changing

Because of the design of our system, the actuation mode changing can be performed only when the robot is stopped. When a clutch is deactivated, the connection between the motor and a fixed reference of the robot is eliminated. Each path ends at zero speed and the robot is supposed to be stable, namely, the PID control has converged. The transition between one actuation mode and another by activating all the clutches and then releasing the unnecessary clutch to the desired actuation mode. When all clutches are activated, the robot becomes over-constrained and no movement of the platform is possible. It is in this state that the data encoders switches it from one mode to the other. The method is to calculate the inverse kinematics introduced previously. First, we store the values of each actuated joint in a vector according to the current actuation mode ϵ_{old} , then we do the same with the new actuation mode ϵ_{new} . The difference between ϵ_{old} and ϵ_{new} sets the offset that is added on the encoder positions.

D. Motions of the robot

When the homing is completed, it is not possible to move the robot in a joint mode. However, the motions of the robot can be realized with two other modes. The first is with the GUI where one can realize translational or rotational movements by pressing buttons up / down of each axis. The condition number of the Jacobian matrix is evaluated before each motion or before each actuation mode changing to prevent the robot from moving near singular configurations. With the second mode, we can load movements that have been generated by another program written in Matlab. We have, for example, a program that reads a G code and generate a file with the position, speed and acceleration sets as well as an integer to define the associated actuation mode. It is thus possible to make a singular assembly mode changing trajectory if the conditions defined in [11] are validated. The video of a path example is given in [15].

V. CONCLUSIONS

In this paper, we have presented the NaVARo robot, a variable actuation robot. With this robot, it is possible to introduce all the problems related to the operating principle as the loss of connection between the encoder and the frames of the robot. A start procedure allows to identify the current assembly mode of the robot by using magnetic sensors and absolute encoders for low accuracy. The movements sensed by the magnetic sensors at low speed and always in the same direction can guarantee the reproducibility of the procedure. A framework is presented to realize the trajectory that does not use the direct kinematics model, which uses only the inverse kinematics model. The redundant sensors ensure that the current method of assembly for the control is the one used by the robot. This allows us to perform assembly mode change paths and confirms that the movement has been achieved. The outstanding work is to improve the location of the platform using the redundancy of the sensors and the calibration of the robot.

ACKNOWLEDGMENT

The work presented in this paper was partially funded by the Erasmus Mundus project India4EU II. Philippe Lemoine is dutifully acknowledged for his technical help in the prototype software debugging.

REFERENCES

- [1] O. Alba-Gomez, P. Wenger and A. Pamanes, *Consistent kinetostatic indices for planar 3-DOF parallel manipulators, application to the optimal kinematic inversion*, In ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 765-774. American Society of Mechanical Engineers, 2005.
- [2] N. Rakotomanga, D. Chablat and S. Caro, *Kinetostatic performance of a planar parallel mechanism with variable actuation*, In Advances in robot kinematics: Analysis and design, pp. 311-320. Springer Netherlands, 2008.
- [3] S. Caro, D. Chablat, P. Wenger and X. Kong, *Kinematic and dynamic modeling of a parallel manipulator with eight actuation modes*, In New Trends in Medical and Service Robots, pp. 315-329. Springer International Publishing, 2014.
- [4] K. Belda and P. Pisa, *Homing, calibration and model-based predictive control for planar parallel robots*, In Proceedings of the UKACC International Conference on Control 2008, 2008.
- [5] M. Palpacelli, M. Callegari, L. Carbonari and G. Palmieri, *Multibody Analysis and Design of a Reconfigurable Parallel Kinematics Manipulator*, In ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. V009T07A076-V009T07A076. American Society of Mechanical Engineers, 2015.
- [6] X. Kong, J. Yu and D. Li, *Reconfiguration Analysis of a 2-DOF 3-4R Parallel Manipulator With Planar Base and Platform*, In ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. V05CT08A034-V05CT08A034. American Society of Mechanical Engineers, 2015.
- [7] W. Ye, Y. Fang and S. Guo, *Reconfigurable parallel mechanisms with planar five-bar metamorphic linkages*, Science China Technological Sciences 57, no. 1 : pp. 210-218, 2014.
- [8] G. Yang, I-M. Chen, W. K. Lee and S. H. Yeo, *Self-calibration of three-legged modular reconfigurable parallel robots based on leg-end distance errors*, Robotica 19, no. 02, pp. 187-198, 2001.
- [9] I-M. Chen, G. Yang, W. K. Lim and S. H. Yeo, *Self-Calibration of Three-Legged Modular Reconfigurable Parallel Robots Based on Measurement Residues*, In Proc. 2nd Intern. Workshop on Computational Kinematics, pp. 117-132, 2002.
- [10] D. Gan, J. S. Dai, J. Dias and L. D. Seneviratne, *Variable Motion/Force Transmissibility of a Metamorphic Parallel Mechanism With Reconfigurable 3T and 3R Motion*, In ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. V05CT08A037-V05CT08A037. American Society of Mechanical Engineers, 2015.
- [11] S. Briot and V. Arakelian, *Optimal force generation in parallel manipulators for passing through the singular position*, The International Journal of Robotics Research 27 (8), pp. 967-983, 2008.
- [12] S. Caro, D. Chablat, and Y. Hu, *Algorithm for the actuation mode selection of the parallel manipulator NAVARO*, In ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. V05BT08A063-V05BT08A063. American Society of Mechanical Engineers, 2014.
- [13] Dspace, *dSPACE - DS1103 PPC Controller Board*, from www.dspace.com/en/pub/home/products/hw/singbord/ppconbo.cfm, Retrieved February 16, 2016.
- [14] C. Gosselin and J-P. Merlet, *On the direct kinematics of planar parallel manipulators: special architectures and number of solutions*, Mechanism and Machine Theory, Vol. 29, No. 8, pp. 1083-1097, 1994.
- [15] NaVARo, *Singular assembly mode changing trajectory*, <https://www.youtube.com/watch?v=e3uMXA2apM>, Retrieved February 16, 2016.